In the Claims:

Please cancel claims 4-9, 14-15, and 19-21, replace claims 1-3, 12-13, 16-18, and 25, and add new claims 26-27, all as shown below.

1.   (Currently Amended):   A <u>high level</u> dynamic code generation method, comprising:

a<u>)</u> creating a class file container object <u>that stores source code describing a class</u>;

b<u>)</u> adding <u>a first source code defining</u> a method to the class <u>stored in the class</u> file <u>container</u> object;

c<u>)</u> adding <u>a second source</u> code <u>into the</u> ~~to~~ method <u>in the class stored in the class file container object</u> ~~using programming language constructs~~;

d<u>) repeating steps b and c to populate the class stored in the class file container object;</u>

e<u>) generating a tree of statements and expressions based on the class stored in the class file container object;</u>

f<u>) using the tree of statements and expressions to generate</u> ~~generating~~ byte code for the class ~~file container object~~; <u>and</u>

g<u>)</u> instantiating an instance of the ~~new~~ class ~~file object;~~

~~generating executable code from the byte code by using a class loader; and~~

~~wherein dynamically generated~~ code would be configured to exist<u>s</u> ~~for the life of a server it resides upon~~.

2.   (Currently Amended):   The method of claim 1 wherein creating a class file container object includes:

~~setting~~ <u>selecting</u> a class name and a super class ~~attributes~~ for a class ~~file~~.

3.   (Currently Amended):   The method of claim 2 wherein ~~the attributes include a~~ any class ~~file~~

name and ~~a parent~~ any super class can be selected.

4 - 9.   (Canceled)

10.   (Previously Presented):   The method of claim 1 wherein the dynamically generated code

implements an adaptor class.

11.   (Previously Presented):   The method of claim 1 wherein the dynamically generated code

implements a proxy class.

12.   (Currently Amended):   The method of claim 1 further comprising:

        repeatedly adding a method to the class stored in the class file container object for each

method associated with a stub generated for a remote object.

13.   (Currently Amended):   The method of claim 12 wherein repeatedly adding a method to the

class stored in the class file container object for each method associated with a stub generated for a

remote object includes:

        determining a number of methods associated with the stub in a remote interface.

14 - 15.   (Canceled)

16.   (Currently Amended):   The method of claim 1 ~~15~~ wherein ~~generating a tree of statements includes:~~

generating a ~~the~~ tree of statements and expressions represents ~~representing~~ at least one method, the at least one method comprising at least one of: a code statement, an expression, a variable and a programming construct.

17.   (Currently Amended):   The method of claim 15 wherein ~~generating a~~ the tree of statements and expressions ~~includes:~~

~~generating a tree forming~~ forms a known structure or interface when the class file container is a known type.

18.   (Currently Amended):   The method of claim 17 wherein ~~generating a~~ the tree of statements and expressions forms ~~forming~~ a known structure or interface when the class ~~file container is a known type includes:~~

~~generating a tree forming a known structure when the class file container~~ is ~~of~~ at least one of an adapter and a proxy type.

19 - 24.   (Canceled)

25.   (Currently Amended): The method of claim 1, wherein the dynamically generated code is used for remote method invocation skeletons, remote method invocation stubs, wrappers for database ~~JDBC~~ connections, and proxies used to enforce call-by-value semantics ~~between EJBs~~.

- 7 -

26.   (New): The method of claim 1, wherein dynamically generated code exists for the life of a server it resides upon.

27.   (New): The method of claim 1, further comprising generating executable code from the byte code by using a class loader.